



DRUID Working Paper No. 05-10

Firm Assets and Investments in Open Source Software Products

By

Andrea Fosfuri, Marco S. Giarratana and Alessandra Luzzi

Danish Research Unit for Industrial Dynamics

www.druid.dk



Firm Assets and Investments in Open Source Software Products

Andrea Fosfuri

Universidad Carlos III de Madrid
Department of Business Administration
Calle Madrid 126, 28903 Getafe (Madrid) - Spain
E-mail: andrea.fosfuri@uc3m.es

Marco S. Giarratana

Universidad Carlos III de Madrid
Department of Business Administration
Calle Madrid 126, 28903 Getafe (Madrid) - Spain
E-mail: marco.giarratana@uc3m.es

Alessandra Luzzi

Universidad Carlos III de Madrid
Department of Business Administration
Calle Madrid 126, 28903 Getafe (Madrid) - Spain
E-mail: alessandra.luzzi@uc3m.es

First Version: March 2005
This Version: July 2005

Abstract:

Open source software (OSS) has recently emerged as a new way to organize innovation and product development in the software industry. This paper investigates the factors that explain the investment of profit-oriented firms in OSS products. Drawing on the resource-based theory of the firm, we focus on the role played by pre-OSS firm assets both upstream and downstream, in the software and the hardware dimensions, to explain the rate of product introduction in OSS. Using a self-assembled database of firms that have announced releases of OSS products during the period 1995-2003, we find that the intensity of product introduction can be explained by a strong position in software technology and downstream market presence in hardware. Firms with consolidated market presence in proprietary software and strong technological competences in hardware are more reluctant to shift to the new paradigm. The evidence is stronger for operating systems than for applications. The fear of cannibalization, the crucial role of absorptive capacity, and complementarities between hardware and software are plausible explanations behind our findings.



Key words: Product Introduction, Open Source Software, Absorptive Capacity
JEL Codes: L860

ISBN 87-7873-174-7

Acknowledgements:

We would like to thank Ashish Arora, and seminar participants at Bocconi University (Milan), IESE Business School (Barcelona), the 12th Product Development Management Conference (Copenhagen) and the 10th DRUID Conference (Copenhagen) for comments and suggestions on an earlier draft. Financial support from *Fundación Ramón Areces* (Andrea Fosfuri and Marco Giarratana), the *Centre Cournot pour la recherche en Economie* (Alessandra Luzzi) and the European Commission (HPSE-CT-2002-00146) is gratefully acknowledged. We also thank Marco Zavarini, software engineer at the SAS Institute, for his guidance in writing the search algorithm for downloading patents and trademarks, and Rebeca Méndez Durón for research assistance. Best Paper Award, 10th DRUID Conference. The usual disclaimers apply.

Introduction

There are no official and fully reliable data on the diffusion of open source software (OSS) products. However, both anecdotal evidence and experts' assessments seem to suggest that OSS development is a phenomenon with a tangible impact on the competitive dynamics of the overall software industry (Lerner and Tirole, 2002).¹

OSS development constitutes a new form to organize innovation and software code writing. Whereas proprietary software production is a rather carefully planned activity supervised by an influential management, OSS production arises from the interactions of loosely related developers organized in a thousand seemingly independent projects (von Hippel and von Krogh, 2003). At the heart of OSS is an attached license, which abandons essential rights granted to the original creator by copyright law. This procedure gives anyone the opportunity to redistribute and modify any received OSS. Most importantly, the source code of the software must be distributed as well or be available at reasonable reproduction cost, although the license must not demand any condition on the software product distributed along with the licensed software code. These characteristics sharply contrast with the existing model of proprietary software, where firms aggressively protect their source codes and use licenses that deprive users of the chance to share and modify the underlying code.

Although OSS development has been around for many years, it is only in the last decade that it has become a sizable phenomenon within the software industry. OSS initiatives have attracted both praise and criticism by software-related firms. On one side, Microsoft has identified OSS as an IPR destroyer. In the words of Jim Allchin, Windows Operating systems Chief Product Manager, "Open-source is an intellectual property destroyer. I can't imagine something worse than this for the software business and the intellectual property business. [...] I don't think we've done enough education of policymakers to understand the threat of open source software" (BusinessWire, 2001). On the other side, companies such as IBM, HP, Sun, Red Hat, Oracle and Novell, have

¹ For instance, GNU/Linux (an open source operating system) has already a consolidated position in the server market. In particular, the Apache web server dominates its product category. A 2004 InformationWeek survey found that 67% of companies use OSS products, with another 16% expecting to use it in 2005; only 17% have no near-term plans to support OSS products. In the browser market, Internet Explorer is losing ground vis-à-vis open source alternatives like Firefox or Opera. In the personal computer operating system, Linux has increased its market share to about 5%.

recently invested in, and legitimized the use of Linux for enterprise applications (Koenig, 2004).

The growing interest in OSS development has not passed unobserved to the economic and management literature. Many scholars have directed their attention to the way in which OSS communities work. In particular a lot of effort has been put in understanding how rewards are assigned within OSS projects and the incentive structure of the developers when the results of individual efforts are immediately available to the public domain (von Hippel and von Krogh, 2003; Lerner and Tirole, 2002). Other research has focused on the stability and the social benefits of the open source model for generating innovation and ideas (Dalle and David, 2005; Gambardella and Hall, 2005). Recent work has looked into the organization of OSS projects and the internal assignment of roles and tasks (Giuri et al. 2004).

This paper takes a rather different approach and investigates the factors that explain the investment of profit-oriented corporations in OSS products. In this respect, we aim at providing new (and most likely first) empirical evidence on two related questions that Lerner and Tirole (2001: 821) have posed: What are the strategies of profit-oriented firms with respect to the OSS movement? And, what is the relationship between OSS business models and the innovative process driven by patents and copyrights? Surprisingly, to the best of our knowledge, there is a lack of empirical research on this facet, and most of the evidence we are aware of is only anecdotal (Lerner and Tirole, 2001 and 2002). This sparse empirical research does not stand for a lack of interest. On the contrary, as we sketched above, the OSS phenomenon starts having profound effects on the software industry and will soon condition firms' strategies and the locus of competitive advantage. In our view, not only it is an issue with a strong practical relevance, but also it represents an important research topic, since investment in OSS products raises the natural question of how firms profit from product introductions in an environment with different rules and conditions. We resort to the resource-based theory of the firm (Barney, 1991; Peteraf, 1993) to assert that differences in the resources available to firms explain different patterns of investment in OSS products.

The underlying theoretical background of this paper is borrowed from the population ecology (Hannan and Freeman, 1984) and the organization behavior (Levinthal and March, 1993) traditions. These scholars posit that, when faced with a new competitive environment, firms are forced to react either through adaptation or

change. Under these circumstances, some resources are critical to guarantee the returns from the investment the firm needs to activate in order to adapt or change, while others might hinder the firm's movement into the new competitive arena.

We focus, in particular, on the role played by downstream (market presence) and upstream (technological competences) firm resources both in the software and hardware dimensions to explain the rate of product introductions in OSS. Indeed, as suggested by Teece (1986), when IPRs are weak and the underlying knowledge base is codified, firms can only rely on the control of complementary downstream assets to guarantee the appropriation of their investment in new products. In addition, since OSS can be viewed as a repository of public-good knowledge, absorptive capacity dictates the firm's ability to identify, assimilate, transform and exploit information that is present outside its own boundaries. In turn, upstream (technological) assets are critical to assess the level of firm absorptive capacity (Cohen and Levinthal, 1990; Ahuja and Katila, 2001; Zahra and George, 2002).

We prove these theories with a self-assembled dataset of firms that have announced releases of OSS products on the specialized press. As mentioned before, although much has been already said about the OSS movement, very little attention has been devoted to understand how profit-oriented firms cope with this new paradigm for software production. Our unique dataset allows us to precisely uncover this lacuna. The dataset is rather comprehensive and tracks down all OSS product introductions from 1995 to 2003, classifying them in two major niches: operating systems and applications. From this source, there are no OSS products released by profit-oriented firms prior to 1995. During the period under scrutiny, we find that 461 different companies have introduced at least one software product based on an OSS scheme.

Our results show that the intensity of product introduction in OSS can be explained by a strong upstream position in proprietary software technology and a significant market presence downstream in hardware. By contrast, firms with strong market presence in proprietary software and good technological competences in hardware are more reluctant to shift to the new paradigm. The evidence is stronger for operating systems than applications. We shall argue below that the fear of cannibalization, the crucial role of absorptive capacity, and complementarities between hardware and software are plausible explanations behind our findings.

The rest of the paper is organized as follows. The next section develops the theoretical framework on which our main hypotheses are based. Section 3 describes the

data, the empirical methodology and the variables used in our regressions. Section 4 shows our results. Section 5 concludes by discussing the implications of our findings, directions for future research and major limitations of our work.

Theory background and hypotheses

Our theory grounds on the idea that pre-existing assets affect the direction and pace of firm change and adaptation (Hannan and Freeman, 1984; Levinthal and March, 1993). As stated in the Introduction, OSS development has recently emerged as a new paradigm in the software industry. Painting it with a broad brush, OSS could be represented as an exogenous shock that hits the industry, and forces firms to respond. Under this scenario, the link between firm resources and competitive achievements is not straightforward. As population ecologists (Hannan and Freeman, 1984; Amburgey et al., 1993) and organization behaviorists (Levinthal and March, 1993; Tripsas and Gavetti, 2000) pointed out, only some of the existing firm assets favor the firm's positioning in a new environment.

Indeed, the actual stock of firm resources is the product of several factors, such as initial endowments, research trajectories, path dependence, that are mostly exogenous to the new conditions (Winter, 1987; Dosi, 1988). Given the new market features, some firms will find easy to adapt and exploit the new opportunities, while others will be forced to activate further investments to acquire the lacking resources. What is more, in some cases the acquisition of new resources could threaten the firm's existing assets and competences (Danneels, 2004). As Gatignon et al. (2002) suggested, adapting mechanisms and the efficacy of time-to-market strategies could be seriously undermined by some existing organizational rigidities generated by the current core competences. Hence, the pre-OSS asset portfolio could affect not only firm ability to extract rents from the new opportunities, but also the ability to organize a reliable business model. This is even more important in a market where the time of entry plays an important role in terms of first mover advantages, reputation building and network externalities.

With this premise in place, in an environment characterized by a low IPRs regime and large availability of public-good knowledge, we focus on co-specialized downstream assets (Teece, 1986) and on absorptive capacity - mapped into upstream

technological assets (Cohen and Levinthal, 1990; Zahra and George, 2002) - as key drivers of firm investments in OSS products.

The seminal work of David Teece represents a milestone to assess the firm's ability to profit from innovation activity (Teece, 1986). A crucial factor to understand such ability is the strength of the protection of intellectual property rights (IPRs). Under tight appropriability regimes, the innovator is more likely to turn its innovative effort into a sustainable competitive advantage. By contrast, under weak appropriability regimes, where the menace of imitation is high, it is the presence of co-specialized assets (services, distribution channels, customers' loyalty, etc.) that allows the firm to protect the flow of rents generated by its innovation activity. "If innovation is not tightly protected [...] then securing the control of complementary capacities is likely to be the key success factor" (Teece, 1986: 37).

In this respect, OSS is the quintessential example of a low IPRs regime. Indeed, under different licensing schemes, open source implies the free access to the software source code, which deprives firms of a classical protection tool of proprietary software, i.e. code secrecy. So, a new OSS product cannot be in itself a source of a sustainable competitive advantage because it is easily imitable (Winter, 1987; Barney, 1991). All in all, this implies that only by focusing on the co-specialized assets we can understand firms' behavior in this new environment. This analysis also echoes the tradition of population ecologists. According to Hannan and Freeman (1984), marketing strategies is one of the most important factors that should be scrutinized to value the ability of firms to adapt in new environments.

Moreover, under a different perspective, the firm's existing knowledge base affects the ability to absorb, transform and utilize external information (Cohen and Levinthal, 1990; Zahra and George, 2002). Van den Bosch et al. (1999) have emphasized how absorptive capacity is the key factor to firm success, especially when most of the valuable knowledge resides in the external environment. Absorptive capacity is even more important in OSS since information is freely available and diffused among the community of users, developers and practitioners. Theoretically, since all firms have access to the same potential pool of external technological knowledge, the ability to learn and utilize it should discriminate the profit opportunities and the corresponding investments. In fact, the extent to which firms leverage the technological information available at the OSS community level may vary significantly according to the structure of the firm's knowledge portfolio (Lenox and King, 2004).

In synthesis, given the salient characteristics of the OSS milieu, we aim at predicting the patterns of firm investments in OSS products through the pre-existing firm endowment of downstream and upstream resources. In so doing, due to the systemic nature of most IT products where hardware and software components are closely intertwined, we distinguish in our analysis between firm assets along these two dimensions. Table 1 summarizes our theoretical framework.

[TABLE 1 ABOUT HERE]

Downstream position in software. Firms that are using profitable business models in proprietary software face a higher risk of product cannibalization by moving to OSS products. Mitchell and Singh (1992) have argued that the fear of product cannibalization prevents firms from investing in emerging and new markets. Typically, profits from new products are uncertain, and the firm is less willing to threaten stable existing rent streams with risky investments (Conner, 1988). In the same vein, the industrial organization tradition has looked at the incentive to innovate of an established monopolist *vis-à-vis* a new entrant. The incumbent has fewer incentives to introduce new products because it also accounts for the erosion of profits in the existing lines of business (Arrow, 1962; Reinganum, 1981). In turn, this implies that the larger is the level of irreversible investments in downstream assets, the higher is the probability of cannibalization and the most likely firms will activate wait-and-see strategies.

On top of cannibalization, existing downstream assets might generate rigidities that make slow and unsuccessful the adjustment towards new business models. A rather common case is a product introduction that requires a modification of the established selling methods. Christensen (1997) points out that when the vendor's systems are not functional to the new markets, the firm's ability to exploit rent opportunities might be seriously undermined. This happens even if the firm owns all the technological resources required. OSS products are usually downloaded directly from customers or uploaded directly by hardware producers. This makes more difficult for firms specialized in proprietary software to adjust their selling practices, typically based on networks of distributors. In a journal interview, the same view was expressed by the research director of Gartner, a consultancy firm specialized in high tech sectors: "Proprietary software vendors are architected to support third-party distribution

channels and cannot compete with direct selling and support models [as in Linux]” (Entrepreneur, 2002: 40).

Moreover, it could be that not only the vendor channels are different but also the type of customers. This is extremely important especially in software, where linkages with users and customers usually increase product performance (software debugging) and help firms in better tailoring products to user needs (Von Hippel, 1998 and 2001). User needs in software are evolving at a rapid pace and the literature has already discussed how the OSS users are intrinsically different from the traditional customers of proprietary software (von Hippel and von Krogh, 2001). OSS buyers are more sophisticated users, sometimes participants of software communities of code compilers. They demand less standardized products with the option to directly fix problems and bugs without any external assistance from post-sale services. Traditional software vendors are not again in a favorable position if the evolution of the market is shaped by this new type of costumers that demand products with a major interaction between the user and the program code. Again Christensen (1997) highlights how firms usually focus and invest in satisfying established costumers in existing markets to whom they made strategic commitments (see also Ghemawat, 1991). This is especially true for software programs that manage money flows and transactions. The crash of such software systems might require immediate problem fixing and therefore the user might ask a direct access to the software code (PCWeek, 1999). User demand of a direct and quick control of the software code creates a new need that is not easily addressed through traditional software vendor systems. These considerations lead to our first hypothesis:

Hypothesis 1: A stronger downstream position in proprietary software will hamper the exploitation of OSS opportunities.

Downstream position in hardware. Firms that own important downstream assets in hardware are in the opposite situation. First of all, absent any severe menace of product cannibalization, they could exploit network externalities generated by the interaction between the software packages and the number of hardware machines sold. The literature has already stressed the positive relationship between software products and hardware machines, being the latter the channel through which network externalities take place. Because of network externalities, the success of a software

product may depend in part on the installed base, especially in the initial phase of product introductions and in the presence of standard battles (Shapiro and Varian, 1998; Brynjolfsson and Kemerer, 1998; Gandal et al., 1999). The history of the success of the DOS paradigm over Macintosh is usually explained by the fact that the largest PC producer, IBM, chose the former as the standard operating systems to provide along with its machines. Therefore, a strong hardware downstream position confers an invaluable advantage to exploit this network effect.

The support of hardware vendors for OSS products might also be interpreted as an attempt to reduce the bargaining power of specialized suppliers of proprietary software. But in our view this is not a sufficient condition, since it does not explain why the hardware sellers should invest in producing their own software products instead of buying cheap OSS packages from specialized firms. Part of this puzzle can be explained by the advantage that hardware vendors have in customizing OSS solutions to their machines, and the overall bundle to the customers' needs. Customization is the avenue to increase product differentiation that usually leads to higher prices and margins (Dewan et al., 2003). The larger is the installed base on which the firm can spread the cost of customization, the higher will be the advantage of this strategy. "One advantage we have [with Linux] is that our solution is very customizable and can be tailored" says Mark Douglas, president of Hyperic, a spin-off of the server producer Covalent Technologies (Computer Reseller News, 2004b). Moreover, Peter Balckmore at H&P Enterprise System Group affirmed: "we will significantly broaden the scope of our Linux offering, providing costumers [...] to take advantage of the low cost, reliability and scalability" (Business Wire, 2002: 448).

Another possible argument could be that hardware producers prefer in-house production because the outside market of specialized OSS vendors is under-developed and it would be too risky in the long run to rely on outsourcing. In the jargon of the industry this is known as "patronizing strategy". By backing OSS initiatives hardware producers ignite the development of OSS products from which they will eventually benefit. The recent acquisition of SUSE by Novell can be reinterpreted in this direction. "We are young and small and Novell's global research, marketing expertise and reputation for security, reliability and global enterprise-level support are exactly what we've been seeking to take SUSE LINUX to the next level" commented Richard Seibt, CEO of SUSE (PR Newswire, 2003).

Granted this, our second hypothesis reads:

Hypothesis 2: A stronger downstream position in hardware will favor the exploitation of OSS opportunities.

Upstream position in software. In an industry characterized by a fast pace of innovation and product-based competition like software, technological capabilities are a key source of competitive advantages (Torrise, 1998). However, the role of technological capabilities takes an interesting twist in OSS. Indeed, in OSS, most of the valuable information resides in the community of users, developers and practitioners that continuously shapes and modifies the nature and the structure of the software projects (von Hippel and von Krogh, 2003). Hence, firm technological capabilities are only marginally employed for the internal generation of new knowledge flows. By contrast, the availability of a large stock of technological capabilities helps not only the acquisition and assimilation of relevant external information but also to build the ability to exploit it (Cohen and Levinthal, 1990). Indeed, mere exposure to external knowledge is not sufficient to internalize it successfully (Pennings and Harianto, 1992).

This is especially important when most of the valuable information resides outside the firm's boundaries. As stated by Van de Bosch et al. (1999), the need to increase firm absorptive capacity is more imperative in highly demanding knowledge environments. Hence, firms in such environments are more likely to exert effort and invest additional resources to increase their level of absorptive capacity (Van de Bosch et al., 1999).

The search and use of new knowledge is a localized and path-dependent process (Winter, 1987; Dosi, 1988; Levinthal and March, 1993). Knowledge accumulation is mostly cumulative, with new knowledge components rooted on the existing knowledge base, long jumps are very rare, and technological competences most often only improve incrementally. Similarly, absorptive capacity is a function of the actual firm knowledge base that influences firm receptivity to seek, recognize and transform external information (Zahra and George 2002; Song et al., 2003). Learning and efficient absorption of external knowledge requires similar cognitive structures, common skills and shared languages. When the existing and the acquired knowledge bases contain elements of similarity, the knowledge integration process, through assimilation and transformation, is much simpler (Ahuja and Katila, 2001). Hence, firms with a large

repository of software competences will be better positioned to benefit from the knowledge available at the level of OSS communities.

This ability to identify, assimilate, transform and exploit external knowledge is especially important for those firms who want to accelerate the time to market of their software products. Since time to market and first mover advantages are often crucial in the first stages of an industry, stronger software technological competences could give firms an additional strategic variable: The ability to speed up the absorption of knowledge residing in the communities of users, developers and practitioners, and to exploit time-based competition.

Hypothesis 3: A stronger position in software technology will favor the exploitation of OSS opportunities.

Upstream position in hardware. Technological competences in hardware have not a clear-cut effect on the firm's ability to exploit OSS opportunities. In supporting this argument, we follow Zahra and George (2002) who suggest that absorptive capacity is a multidimensional construct that impinges at different times on different capabilities and routines. Specifically, they point out the existence of two sub-sets of absorptive capacity: potential and realized. Potential absorptive capacity enables a firm's receptiveness to external knowledge, and defines the boundaries to the amount of external information the firm can identify and assimilate; realized absorptive capacity reflects a firm's ability to leverage absorbed knowledge and transform it into innovation outcome. A similar characterization was advocated by Arora and Gambardella (1994), who distinguished between the ability to evaluate external knowledge and the ability to exploit it, and it is also implicit in Cohen and Levinthal (1990)'s contribution.

We argue below that the relationship between hardware competences and absorptive capacity can only be understood by analyzing separately potential and realized absorptive capacity. Indeed, on the one hand, hardware competences could generate positive technological synergies with the software component thereby improving the firm's ability to transform and exploit externally available software knowledge (i.e. its realized absorptive capacity); on the other hand, hardware competences could harm the firm's ability to identify and assimilate external software information (i.e. its potential absorptive capacity). The net effect is therefore unclear.

Indeed, a deeper understanding of the mechanisms of a hardware machine could help build up better algorithms and software architectures, using at best the information coming from the environment. Hardware competences should allow “encapsulating hardware and software resources in a unique component” (Electronic Engineering Time, 1999). Put it differently, a better understanding of the machine can be exploited through software products that closely match the idiosyncrasies of the underlying hardware.

However, the relatedness of the knowledge base is an important factor to set the boundaries of the amount of external information the firm is able to scan and absorb (Ahuja and Katila 2001). Hardware competences are more distant to the potential pool of information available in the OSS communities compared to software competences. Moreover, there exists a big difference in terms of technological languages between hardware and software engineers. The specialized press usually refers to a hardware-software trade-off to capture this difference (Electronic Engineering Time, 1999). Strong hardware background could sometimes generate difficulties in coping with the fast changing best-practices in software code writing (Kasser and Shoshany, 2000).

In sum, hardware competences can help writing software that increases the quality of the final joint product, thanks to the hardware-software co-design, but they lower the firm’s ability to identify and assimilate valuable information available in the OSS community. This is particularly true for some software niches where hardware components and software code quality are critical, such as videogames (Computer Graphics World, 2004).

Hypothesis 4: A stronger position in hardware technology has an ambiguous effect on the exploitation of OSS opportunities.

Data and methodology

The sample

Our sample is composed by the firms that have announced on the specialized press the introduction of a software package based on Open Source architectures between 1980 and 2003. Product introduction data were taken from Infotrac's General Business File ASAP and PROMT databases that, from a large set of trade journals, magazines and other specialized press, report several categories of “events” classified by industrial sectors. We searched for all press articles that reported a “Product announcement”, a “New software release” and a “Software evaluation” in the software sector (SIC Code

7372, software) and that contained in the text of the article the words “open source” or “linux”. After carefully controlling that we were only picking up OSS products, we extracted from each article the name of the company and the date of product introduction (month and year). We found that the first announcement of an OSS product was in May 1995.

Using the information on the SIC codes, we divided the OSS products in two main niches: the operating systems, OPSYS (SIC Codes 737261 and 737250) and the applications, APP (6 digit SIC Codes other than -61 and -50). When the SIC code was only available at 4 digits, the text of the article was carefully inspected in order to distinguish between the two niches. To give an example, the SUSE Linux was classified under the OPSYS category while the Matlab version for Linux was classified in APP. This distinction is quite important since we want to track potential differences in the drivers of firm investment in the two niches. As a matter of fact, the literature (Gandal et al., 1999) has stressed that success in operating systems is more directly linked to hardware synergies and network effects, while for applications the key variables are innovation and software efficiency in accomplishing the assigned tasks.

We controlled for sample firm group structure using information from the Business and Company Resource Center database, Gale Group's Infotrac. We ended up with 213 different entrants in the APP niche, which introduced 360 different products, and 320 different entrants in the OPSYS niche, which introduced 877 different products. Since 72 firms entered both niches, the total number of entrants in the OSS market is 461.² The firm level data used in our analysis were recovered from Bureau van Dijk's Icarus, Amadeus and Jade databases, for American, European and Asian firms respectively. Table 2 reports our sample of firms classified according to their core sector of activity.

[TABLE 2 ABOUT HERE]

Empirical model

² Entry is here defined using the announcement of an OSS product. Of course, there can be cases in which the firm has announced a product but it was not afterwards commercialized or, although the product was introduced, it did not attract demand and was immediately shelved. This might end up introducing some measurement problem in our dependent variable. Unless such measurement error is correlated with our explanatory variables, it would not generate serious concerns.

The sample described before suffers from a sample selection problem because we do not observe firms that did not introduce any OSS product before December 2003. To control for this problem, we built up a control sample. In particular, firms belonging to the control group were randomly selected from Bureau Van Dijk's Icarus for North American firms, Amadeus for European firms and Jade for Asian firms, using the precise country (Japan, Germany, UK, Canada ...) and sector (3 digit SIC code) composition of the original sample as matching criteria between cases and controls. The total dimension of the original sample was replicated for the control sample in both the APP and OPSYS niches.

Conditional on the fact that a firm actually introduced a product in OSS, the number of product introductions can then be estimated by nonlinear least squares (Terza, 1995). Specifically, considering both the firms in the original sample and in the control group, we assume that the observed number of product introductions in OSS, y_i , follows a Poisson distribution conditional on an error term that is normally distributed: $y_i | \varepsilon_i \sim \text{Poisson}(\lambda_i)$, where $\varepsilon_i \sim N(0, \sigma^2)$. Its log-mean can thus be modeled as a linear function of our covariates and controls, x_i , plus the error term ε_i :

$$\ln E(y_i | x_i, \varepsilon_i) = \ln \lambda_i | \varepsilon_i = \beta' x_i + \varepsilon_i.$$

The normal error term in this specification allows taking into account unobserved heterogeneity, which is not modeled in the standard Poisson specification where the mean of the distribution of y_i is equal to the variance and all of the heterogeneity is accounted for in the vector of covariates x_i ³.

We then address the sample selection through a Probit specification where our dependent variable is a dummy equal to 1 if the firm actually introduced an OSS product and 0 otherwise. This choice can be modeled as:

$$d_i = \begin{cases} 1 & \text{if } z_i^* = \alpha' w_i + u_i > 0 \\ 0 & \text{otherwise} \end{cases}$$

where w_i are the covariates used to explain the decision to enter or to not enter with a an OSS product, $u_i \sim N(0,1)$ and the two error terms (ε_i, u_i) have a bivariate normal

³ The more common way to model heterogeneity in the Poisson model is to re-specify the unconditional distribution for y_i as a negative binomial (hypothesizing the error term ε_i in the conditional Poisson distribution of $y_i | \varepsilon_i$ to have a gamma $(1, \xi)$ distribution). Unfortunately, this formulation does not permit to address the sample-selection problem (Greene, 1997).

distribution with covariance $\theta = \rho\sigma$. The correlation term ρ between the errors in the two specifications allows us to take into account unobserved factors that may influence both the decision to enter with a product and the decision of how many products to launch, thus solving the selectivity problem.

Under this framework, the mean of the number of product introductions conditional on $d_i = 1$ can be specified as (see Terza, 1995):

$$E[y_i | x_i, d_i = 1] = \exp(\beta^0 + \beta_1 x_{1i} + \dots) \left(\frac{\Phi(\alpha' w_i + \theta)}{\Phi(\alpha' w_i)} \right),$$

where Φ is the cumulative function of a standardized normal distribution.⁴ Notice that, even if the actual distribution of $y_i | x_i, d_i = 1$ is unknown, the coefficients in the conditional mean function can be estimated by non-linear least squares. Consistent estimates are obtained estimating the full model with a two-steps procedure, where the parameters α in the Probit specification are first estimated on the whole number of observations in the sample and in the control sample. Then, the parameters β and θ are estimated with least squares on the sub-sample of firms that actually introduced a product. We use a heteroskedasticity-consistent estimator for the asymptotic covariance matrix (Murphy and Topel, 1985) of the estimates of the second step.

To control for potential differences, we run two separated estimations for the two niches: APP and OPSYS.

Dependent variable

Our unit of observation is the firm. The dependent variable in the second step estimation is the number of new OSS packages (or new versions of already launched OSS products) announced by the firm from its first OSS product to December 2003. This variable aims, first, at capturing the intensity of firm investment in OSS products. All things being equal, the number of products launched, N_i , is usually directly influenced by the level of investment: $N_i = f(I_i)$ with $f' > 0$. The literature has shown that in a patent production function the number of patented innovations is highly correlated to the investment in R&D (Hausman et al., 1984). Second, since we are analyzing firms' investments at the beginning of a new paradigm, the intensity of product introductions could also signal the intention of the firm to “rush” into the market in order to benefit

⁴ The constant term β^0 estimated in this specification is actually equal to $\beta_0 + \sigma^2 / 2$.

from lead time, first mover advantages, reputation building and cycle time. In this respect, the literature has pointed out that when analyzing time to market we should acknowledge first the firm's recognition of potential time advantages (and not all firms have it) and then its ability to organize the production to exploit them (Harter et al., 2000).

Independent variables and controls

Firm upstream positions in software and hardware are measured through the firm stock of patents. This is a rather common and well-accepted indicator employed in the innovation literature. For instance, several works both from the industrial organization and the management traditions have used firm patent stock to proxy for R&D cumulative expertise and knowledge, technological ability and absorptive capacity (Henderson and Cockburn, 1993; Arora and Gambardella, 1994; Silverman, 1999; McGrath and Nerkar, 2004). "A patent, by definition, represents a unique and novel element of knowledge. A set of patents then represents a collection of discrete, distinct units of knowledge. Identifying a set of patents that a firm has demonstrated familiarity with [...] can be a basis for identifying the revealed knowledge base of a firm" (Ahuja and Katila 2001: 201). Specifically, we proxy the firm's pre-entry technological assets in software and hardware with the stock of patents granted in these two technological fields at the US Patent and Trademark Office the month before the firm's announcement of its first OSS product. We label these two covariates: PATENTSOFT and PATENTHARD. In order to classify patents in the two categories, we utilized search algorithms in the abstract of the patent. We report in the Appendix the detailed explanation of these search algorithms.

We apply a similar procedure to measure firm pre-entry downstream assets. We utilize the firm's "live" stock of trademarks filed at USPTO the month before the firm's first announcement of OSS product introduction. Trademarks are a good proxy for the firm's downstream position. With a sample of 237 COMPUSTAT firms from the period 1995-1998, Seethamraju (2003) reports a highly significant empirical association between trademarks and sales. Trademarks are combinations of "words, phrases, symbols or designs that identify and distinguish the source of the goods or services of

one party from those of others” (USPTO Documentation, <http://tess.uspto.gov>).⁵ Firms can register as a trademark a new brand, a jingle or a slogan, a new image or a logo. Trademarks are extremely important especially in software. An emblematic case is the court battle that Microsoft has filed against Lindows.com for the use of the trademarks “Lindows”, “Lindows.com” and “LindowsOS” that according to Microsoft infringed several Microsoft trademarks. The court not only agreed in prohibiting the use of these trademarks but also assigned a penalty of about 350,000 Euros (PR Newswire, 2003).

In order to distinguish between software and hardware trademarks we have applied to the front page of the trademark search algorithms similar to the ones used for patents (see the Appendix). We then created two variables: TRADEMARKSOFT and TRADEMARKHARD, i.e. the pre-entry number of “live” trademarks filed by the firm up to the month before its first announcement of an OSS product in software and in hardware respectively.

For the firms in the control sample, we take trademark and patent stocks at December 2003.

We also consider a set of control variables in our estimations. First, we want to account for the fact that firms that have entered the OSS market earlier have more time to launch products during the sample period. TIME2MKT is thus defined as the number of months that elapsed between the firm’s entry (i.e. its first OSS product) and the end of the period, December 2003. Even if, by definition, this variable is highly correlated with the dependent variable, our aim is to test the significance of the covariates of theoretical interest beyond the explanatory power of TIME2MKT.

Second, we control for firm size and age. EMPL is the number of firm employees and AGE is the firm age (measured in years) at the date of the first OSS product announcement. Data were taken from Bureau van Dijk’s Icarus, Amadeus and Jade. These are standard controls in the entry and survival literature (Klepper, 2002), and could be interpreted as proxies for firm scale and experience. Size is especially important to scale down the stock of firm assets (patents and trademarks).

Third, to control for core sector differences, we introduce five dummies that group firms with the same core businesses. The sector dummies are HARDWARE (SIC code 357), SOFTWARE (737), ELECTRONICS (36 except for 367 and 366),

⁵ The US trademark owners pay different types of fees for each class of goods/services for which a trademark is registered, and they have to prove periodically that they are using the trademark in the US market, otherwise the trademark is cancelled (i.e. it is not longer “live”).

SEMICONDUCTORS (367) and TELECOMMUNICATIONS (366). Firm core sectors were similarly taken from Bureau van Dijk's databases (3 digit SIC codes). Hardware and software dummies are especially important because the effect of upstream and downstream software and hardware assets will be potentially significant beyond these controls, i.e. taken two hardware firms, the one with more trademarks in hardware will invest more in OSS opportunities.

Fourth, we introduce two geographical dummies, EUROPE for the European firms, and ASIA for the Asian firms. The baseline is an American firm.

Finally, since we run separate estimations for the two niches (APP and OPSYST) and 72 firms announced product introductions in both, we want to control for the fact that firms that have already entered one niche might have a substantially different rate of product releases in the other niche. Therefore, when we estimate our regression for the OPSYST niche, we introduce a dummy that equals 1 if the firm has introduced a software application before the introduction of the first operating system, and zero otherwise. This variable is labeled APPBEFORE. Similarly, for the APP niche, we create OPSYSTBEFORE, a dummy variable that takes the value 1 if the firm has introduced a product in the operating systems niche before the first product introduction in applications. The logic of these dummies grounds on previous studies that have found positive and strong feedbacks between applications and operating systems. Comparing DOS and CP/M products, Gandal et al. (1999) highlight that applications make operating systems more valuable, and that the reverse relationship also holds (operating systems increase the value of applications). Following these findings, a firm that has entered in both niches should benefit of some sort of advantage in releasing new software products. Moreover, these two dummies represent our best proxies to capture for potential product bundling strategies, since we do not have any information on whether these products are sold in a unique package or separately.

The variables measuring firm pre-entry technological and downstream assets were used in both the first and the second stage estimation. The same was done for our measure of size and for sector and country dummies. The variables OPSYSTBEFORE, APPSYSTBEFORE and TIME2MKT were obviously considered only in the second step estimation on the sub-sample of firms that entered the OSS market. Finally, the variable AGE was used only to explain firm decision to enter with a product.⁶

⁶ We do not have any strong theory on the covariates of the first step estimation that we interpret only as a means to control for potential sample selection problems. The econometric theory suggests that it is

Table 3 shows the basic descriptive statistics for our independent variables and controls in the two niches for the sample and the control sample of firms.

[TABLE 3 ABOUT HERE]

Results

Table 4 shows the results of our estimations. In particular, this table reports the estimation results of the second step non-linear least squares regression, that is, the effect of our covariates on the frequency of product introductions based on OSS architectures. Figure 1 and 2 depict the number of predicted product introductions according to the variations from the minimum to the maximum of our core variables, all other variables held at their mean values.

First, we discuss the effect of firm pre-entry assets on the number of product introductions in the operating systems niche. Next, we show how substantial differences can be observed for the applications niche, and provide a tentative explanation for this finding.

First of all, notice that most of the covariates of theoretical interest show up a significant effect.

In accordance with our first hypothesis, the stronger the downstream position in proprietary software, the less the firm will invest in OSS operating systems. Though significant at 1% level, the negative effect of the pre-entry stock of software trademarks on the number of product introductions is low in magnitude, as the model prediction in figure 1.a shows.

The effect of a stronger downstream position in hardware goes in the opposite direction, showing a positive and significant effect on the number of product introductions in the OPSYS niche. Contrary to the previous case, the magnitude of the effect is more relevant. This finding conforms to our hypothesis 2.

The pre-entry position in software technology also enhances the exploitation of OSS opportunities, supporting our third hypothesis. In the fourth hypothesis we predicted an ambiguous effect of pre-entry technological competences in hardware. The results highlight that these assets slow down the frequency of product introductions, but once again with a small effect in terms of magnitude (figure 1.b). These latter findings

advisable not to perfectly replicate the covariates in the two steps, including in the first step covariates that are not used in the second (Vella, 1998; Terza, 1998). We therefore follow this indication.

support the conjecture that absorptive capacity is crucial to identify, assimilate, transform and exploit the repository of knowledge available at the level of OSS communities. In this respect, while firm software resources clearly favor both the potential and realized absorptive capacity, hardware knowledge has a negative impact on the former and a positive on the latter. Theoretically, the net effect is ambiguous. Empirically, our results show that the negative impact on potential absorptive capacity prevails. We think that the strong positive sign of PATENTSOFT and the slightly negative coefficient of PATENTHARD are in line with this interpretation.

Apart from the negative impact of the pre-entry stock of software trademarks, firm pre-entry assets do not seem to have a significant role on the frequency of product introduction in the application niche.⁷ Firm pre-entry position in hardware shows an effect only in a model without the OPSYSBEFORE control (Table 4-(2)). The factor that seems to explain most of the intensity of product launching in the APP niche is indeed whether the firm had previously entered the operating systems niche or not. As illustrated in Figure 2, firms that have entered the operating systems niche before launching the first OSS product in applications show a higher release rate in this niche.

These findings suggest that in the APP niche firms that have already introduced operating system products are more active players. This offers a rather coherent picture of the overall firm's behavior: In the operating systems the pre-entry assets are clear-cut drivers of the firm's incentives to invest (or not) in OSS opportunities, while in the applications the role of firm assets is shadowed by firm previous entry in the operating systems niche. This picture is consistent with an initial stage of the OSS market, where - lacking an established and reliable set of application producers - firms that want to launch operating systems are in some way "forced" to release in-house applications in order to increase the value of their OSS operating systems. Earlier studies on the software industry (see for instance, Gandal et al., 1999) have demonstrated how the success of an operating systems paradigm is intrinsically associated with the availability of a large portfolio of applications that could be run on that software base. This recalls the patronizing strategies that some firms pursue: By backing OSS applications they increase the value of OSS operating systems. This vision is confirmed by the words of Rich Severa, president of MOCA, a Sun Microsystems' division focused on OSS

⁷ Notice, however, that the signs of our core covariates go all in the correct direction, although given the high standard errors they are not significant. For the APP niche we have a smaller number of observations, which might reduce the precision of our estimates.

products: “The greater value-added your software product has, as perceived by your potential customers, the more difficult it is for competitors to win business away from you. Accordingly, one of the best ways to create and sustain a profitable business is to be focused on your value-added rather than your product. When potential customers readily see the additional value in what you offer in terms of meeting their needs, it is easier to make the sale while keeping your prices and profits higher. With Linux we're not looking to jump into a new segment of customer that wants to buy everything. If we can add more opportunities to end users with patronizing, that's fine” (Computer Reseller News, 2002).

Concerning our main control covariates, TIME2MKT is significant as expected, suggesting that the firm that have entered before launch more OSS products. Also SIZE is significant and positive, indicating that a scale effect is at work here.

Discussion and Conclusions

Drawing on the resource-based theory of the firm (Barney, 1991; Peteraf, 1993), this paper has investigated which assets explain the rate of product introductions by profit-oriented organizations in OSS. OSS development constitutes a new form to organize innovation and software code writing, whose impact on the overall software industry has become particularly sizable in the second half of the 90s. Mirrored in a new competitive scenario, firms have been forced to react through adaptation and change. Following a consolidated tradition both from organization behavior (Lenvinthal and March 1993; Tripsas and Gavetti 2000) and population ecology (Hannan and Freeman 1984), we have argued that the pre-OSS firm assets play a crucial role in shaping the incentives and the ability to make inroads into the new competitive arena.

OSS has two key features, which we have tried to accommodate within our theory. First, it is *par excellence* an environment with weak IPRs. This, in addition to the highly codified nature of software knowledge, makes appropriability of innovation efforts an important concern for firms (Teece, 1986). Second, OSS communities can be viewed as repositories of public-good knowledge. Since everybody is entitled to access and use such external stock of knowledge, it is the ability to identify, assimilate, transform and exploit it that generates heterogeneity across firms. Absorptive capacity is therefore a key source of competitive advantages in OSS (Cohen and Levinthal, 1990). These two features of OSS have motivated our focus on downstream and upstream (technological) assets as major drivers of OSS product introductions.

This paper contributes to the literature on innovation management and absorptive capacity in several ways.

First, our results point out to the crucial role of complementary downstream assets as drivers of firm investment in OSS. Indeed, we show that firms with a strong position in the software market are more reluctant to embrace the new paradigm, while firms with considerable stakes in the hardware market have a very positive attitude towards OSS products. These findings suggest that in a regime of weak IPRs, complementarities with other firm assets are key to guarantee the appropriability of innovation efforts (Teece, 1986). Hardware producers can customize OSS products to their machines, and the overall bundle to their customers' needs. The investment in OSS allows them to increase the value-added of their offerings (Dewan et al., 2003). By contrast, established leaders in the software industry have clear difficulties to grasp OSS opportunities. As suggested by several scholars, incumbents tend to react slowly and late to paradigm shifts and drastic technological changes (Tushman and Anderson, 1986). The literature has highlighted several explanations that go from incentives (Arrow, 1962; Reinganum, 1981) to important organizational reshaping (Henderson and Clark, 1990), from inertia (Tripsas and Gavetti, 2000) to established customers networks (Christensen, 1997). Most of them are present in the transition from proprietary software to OSS production. Interesting enough, this also confirms the serious potential threat that the OSS movement exerts on consolidated leaders in the software industry. As the theory predicts, they have been slow in embracing the new paradigm. The extent to which OSS production will account for a considerable share of all software products is unknown, unpredictable and out of the scope of this paper. However, our results point out that if OSS will take off there will be a considerable reshaping of competitive positions in the overall software industry.

Second, our research reveals that technological assets are also crucial drivers of OSS product introductions. We interpret these findings as evidence of the importance of absorptive capacity (Cohen and Levinthal, 1990). However, not all technological competences have the same impact on the firm's level of absorptive capacity. We draw on Zahra and George (2002)'s distinction between potential and realized absorptive capacity, to argue that while software competences help increase both sets of absorptive capacity, hardware competences might correlate negatively to potential absorptive capacity. Indeed, hardware and software engineers use different languages and do not share similar knowledge bases. This makes the identification and assimilation of

external knowledge much more difficult (Ahuja and Katila, 2001). The key role of absorptive capacity to understand progresses and trends within the OSS communities has also practical relevance for firms attempting to cope with the new paradigm for software production. Indeed, previous research has shown that proprietary software firms allow their engineers to contribute to the OSS communities as a means to monitor external knowledge development and to build the necessary absorptive capacity to exploit potential opportunities (Lerner and Tirole, 2002).

Third, our paper also contributes to the literature on the dynamics of industries characterized by network externalities (Shapiro and Varian, 1998; Gandal et al., 1999). Operating systems are especially subject to strong network effects. The larger the number of users adopting a given operating system, the stronger the interest of application suppliers to develop software products especially tailored to that operating system. In turn, a larger portfolio of applications makes the operating system more valuable to consumers. Indeed, we find that a strong downstream position in hardware is positively associated to the rate of OSS production introductions in operating systems. The literature has stressed that the relationship between the software products and the hardware machines is the channel through which network externalities take place (Shapiro and Varian, 1998; Brynjolfsson and Kemerer, 1998). In addition, our findings reveal that in the applications niche firms that have previously released OSS operating systems show a higher intensity of product introduction. This result is consistent with an early phase of the OSS market, where, lacking a structured supply of OSS application producers, the firms that have launched OSS operating systems show higher incentives to release in-house applications in order to increase the value of their operating systems.

Our contribution in terms of measurements is also worth mentioning. The recent literature has provided several exciting ways of using patent data to measure different constructs of knowledge (Henderson and Cockburn, 1994; Jaffe et al., 1993; Stuart and Podolny, 1996). Indeed, in this study we used patent accounts to measure the firm's stocks of knowledge in hardware and software. What is more, a real novel contribution, we use another large public available data, the US Trademarks dataset, to measure firms' market positions (Mendoca et al., 2004). Previous research has shown that the stock of "live" trademarks is highly correlated with the firm's market share (Seethamraju, 2003). A virtue of the measure used in this paper is that it is easily available, and allows, by using information on individual elements of trademarks

portfolios, to make a fine-grained assessment of a firm's market position. We indeed exploit this information to distinguish between hardware and software.

Finally, to the best of our knowledge, this is the first paper that provides some comprehensive empirical evidence on how profit-oriented organizations cope with the new paradigm for software production. It goes without saying that there are many interesting aspects that we were unable to uncover during this research. These are limitations of the present study and natural candidates for future research in this area. Specifically, we do not have any measure of the level of openness that is attached to the software packages our sample firms have released. We can only trust the fact that the firms announced on the specialized press, and to all the OSS community, a release of a software product that is claimed to be "open source". The degree of openness would be an important piece of information to understand how firms defend their innovative efforts, and profit from OSS products. Related, we do not have information about the type of license under which the software product is sold. OSS product can typically be modified and resold. However, restrictions are often imposed by the license agreement. Such information, when available, could cast new light on the business model and commercialization strategies chosen to profit from OSS products. It will also be useful to understand the interaction between the firms and the open community of users, practitioners and developers. Finally, a direct measure of the performance of OSS products, like sales or returns to investment, will be useful to value the actual potential of the firms' strategies, and to carefully disentangle OSS product introductions that are successful from those that fail.

References

- Ahuja G., Katila R. (2001). Technological Acquisitions and the Innovation Performance of Acquiring Firms: A Longitudinal Study *Strategic Management Journal* 22 (3): 197-220.
- Amburgey T.L., Kelly D., Barnett W.P. (1993). Resetting the Clock. The Dynamics of Organizational Change and Failure. *Administrative Science Quarterly* 38 (1): 51-73.
- Arora A., Gambardella A. (1994) Evaluating Technological Information and Utilizing It. Scientific Knowledge, Technological Capability, and External Linkages in Biotechnology. *Journal of Economic Behaviour and Organization* 24 (1): 91-114.
- Arrow K.J. (1962). Economic Welfare and the Allocations Resources for Inventions, in *The Rate and The Direction of Inventive Activities*, Nelson R.R. (Ed.) Princeton NJ: NBER Princeton University Press.
- Barney J. (1991). Firm Resource and Sustained Competitive Advantage. *Journal of Management* 17 (1): 99-120.
- Bessen J., Hunt R.M. (2004). An Empirical Look at Software Patents. Working Paper 30/17 Federal Reserve NY.
- Bresnahan T.F., Greenstein S. (1999). Technological Competition and the Structure of the Computer Industry. *Journal of Industrial Economics* 47 (1): 1-40.
- Brynjolfsson E., Kemerer C.F. (1996). Network Externalities in Microcomputer Software: An Econometric Analysis of the Spreadsheet Market. *Management Science* 42 (12): 1627-1647.
- BusinessWire. (2001). Embedded Linux Consortium Claims Microsoft Position Ignores Vast Embedded Markets. 14 (5): 105 (7/5/2001).
- BusinessWire. (2002). HP and Red Hat Expand Relationship to Deliver Linux Solutions to Enterprise Customers. 15 (2): 448 (18/6/2002).
- Christensen C. (1997). *The Innovator's Dilemma*. Boston: Harvard Business School Press.
- Cohen W., Levinthal D. (1990). Absorptive capacity: A new perspective on learning and innovation. *Administrative Science Quarterly* 35(1): 128-152.
- Computer Graphics World. (2004). Fear Factors: Game Developers Use the Latest Tools and Techniques to Create Frightening Experiences. 27 (12): 26-32 (December).

- Computer Reseller News. (2002). MOCA Stands Firm. Distributor says it will keep focus on enterprise as Sun eyes SMBs. 14 (9): 74 (30/9/2002).
- Computer Reseller News. (2004a). SOA = MBA. The Advent of Services-Oriented Architectures is Paving the Way for Solution Providers to be at the Vanguard of Fulfilling the Long-Awaited Promise of Melding Business Processes with IT. 16 (8): 18 (9/8/2004).
- Computer Reseller News. (2004b). Hyperic Eases Network Tension. New Platform Manages Web Apps Infrastructure. 16 (7): 29 (12/7/2004).
- Conner K.R. (1998). Strategies for Product Cannibalism. *Strategic Management Journal* 9: 9-26.
- Dalle J.M., David P.A. (2005). SimCode: Agent-based Simulation Modelling of Open-Source Software Development, in *The Economics of the Internet*, Curien N. and Brousseau E. (eds.), Cambridge: Cambridge U.P.
- Danneels E. (2004). Disruptive Technology Reconsidered: A Critique and Research Agenda. *Journal of Product Innovation Management* 21 (4): 246-258.
- Dewan R., Jing B., Seidmann A. (2003). Product Customization and Price Competition on the Internet. *Management Science* 49 (8): 1055-1070.
- Dosi G. (1988). Sources, Procedures and Microeconomic Effects of Innovation. *Journal of Economic Literature* 26 (3): 1120-1171.
- Electronic Design. (2002). Programmable Logic: To HDL Or not to HDL? Engineers Grappling with FPGA Design have New EDA Tools Ready to Reclaim Time-to-Design Completion. 50 (27): 43-47 (23/12/2002).
- Electronic Engineering Times. (1999). Components Simplify DSP Design. 13 (11): 108 (1/11/1999).
- Electronic Engineering Times. (2000). Language: Hard/Software Trade-offs. 14 (9): 110 (25/9/2000).
- Electronic News. (1999). The Softer Side of Test: Software Products to Star at International Test Conference. 45 (9): 24 (27/9/1999).
- Entrepreneur. (2001). The Penguin's Revenge. 29 (7): 40 (July).
- eWeek (2004). EclipseCon: IBM, Sun Set the Tools Stage 8 (2): 11 (2/2/2004).
- Gambardella A., Hall B.H. (2005). Proprietary vs Public Domain Licensing of Software and Research Products. NBER Working Paper w11120.
- Gandal N. (2001). The Dynamics of Competition in the Internet Search Engine Market. *International Journal of Industrial Organization* 19 (7): 1103-1117.

- Gandal N., Greenstein S., Salant D. (1999). Adoptions and Orphans in the Early Microcomputer Market. *Journal of Industrial Economics* 47 (1): 87-105.
- Gatignon H., Tushman M.L., Smith W., Anderson P. (2002). A Structural Approach to Assessing Innovation: Construct Development of Innovation Locus, Type, and Characteristics. *Management Science* 48 (9): 1103-1122.
- Ghemawat P. (1991). Market Incumbency and Technological Inertia. *Marketing Science* 10 (2): 161-173.
- Giuri P., Ploner M., Rullani F., Torrisi S. (2004). Skills and Openness of OSS Projects: Implications for Performance. LEM Working Paper 2004/19.
- Hannan M., Freeman J. (1984). Structural Inertia and Organization Change. *American Sociological Review* 49 (2): 149-164.
- Hausman J., Hall B. H., Griliches Z. (1984). Econometric Models for Count Data with an Application to the Patents R&D Relationship. *Econometrica* 52 (4): 909-938.
- Henderson R. (1993). Underinvestment and incompetence as responses to radical innovation: evidence from the photolithographic alignment equipment industry. *RAND Journal of Economics* 24(2): 248-270.
- Henderson R., Clark K. (1990). Architectural innovation: the reconfiguration of existing product technologies and the failure of established firms. *Administrative Science Quarterly* 35(1): 9-30.
- Henderson R., Cockburn I. (1994). Measuring competence? Exploring firm effects in pharmaceutical research. *Strategic Management Journal* 15: 63-84.
- Kasser J.E., Shoshany S. (2000). Systems Engineers are from Mars, Software Engineers are from Venus, Thirteenth International Conference "Software & Systems Engineering & their Applications", Paris (December).
- Klepper S. (2002). Firm survival and the evolution of oligopoly. *Rand Journal of Economics* 33(1) 37-61.
- Koenig J. (2004). Seven open source business strategies for competitive advantage. *IT Manager's Journal*. May 14.
- Lenox M, King A. (2004). Prospects for Developing Absorptive Capacity Through Internal Information Provision. *Strategic Management Journal* 25(4): 331-345
- Lerner J., Tirole J. (2001). The Open Source Movement: Key Research Questions. *European Economic Review* 45(2): 819-826.
- Lerner J., Tirole J. (2002). Some Simple Economics of Open Source. *Journal of Industrial Economics* 50(2): 197-234.

- Levinthal D., March J. (1993). The Myopia of Learning. *Strategic Management Journal* 14: 95-112.
- McGrath R.G., Nerkar A. (2004) Real Options Reasoning and a New Look at the R&D Investment Strategies of Pharmaceutical Firms. *Strategic Management Journal* 25(1): 1-21
- Mendonca S., Pereira T.S., Godinho M.M. (2004). Trademarks as an Indicator of Innovation and Industrial Change. *Research Policy* 33 (9): 1385-1404.
- Mitchell W., Singh K. (1992). Incumbents Use of Preentry Alliances Before Expansion into New Technical Subfields of an Industry. *Journal of Economic Behavior and Organization* 18(3): 347-372.
- Pennings J.M., Harianto F. (1992). The Diffusion of Technological Innovation in the Commercial Banking Industry. *Strategic Management Journal* 13: 29-46.
- Peteraf M.A. (1993). The Cornerstones of Competitive Advantage: A Resource-Based View. *Strategic Management Journal* 14: 179-191.
- PC Week. (1999). Testing the Linux Waters. 16 (11): 81-82 (15/3/1999).
- PR Newswire (2003). Novell Announces Agreement to Acquire Leading Enterprise Linux Technology Company SUSE 4/11/2005.
- PR Newswire. (2003). Microsoft Blocks Distribution of Desktop Linux in Sweden; Judge Issues Temporary Restraining Order Halting LindowsOS Sales to Swedes. 7(12): 5 (11/12/2003).
- Reinganum J.F. (1981). Market Structure and the Diffusion of New Technology. *Bell Journal of Economics* 12(2): 618-624.
- Seethamraju C. (2003). The Value Relevance of Trademarks, in *Intangible Assets. Values, Measures and Risks*, Hand, J.R.M. and Lev, B. (eds), Oxford: Oxford University Press.
- Shapiro C., Varian H. (1998). *Information Rules: A Strategic Guide to the Network Economy*. Boston: Harvard Business School Press.
- Silverman B.S. (1999). Technological Resources and the Direction of Corporate Diversification: Toward an Integration of the Resource-based View and Transaction Cost Economics *Management Science* 45 (8): 1109-1124.
- Smith R. (1999). *Internet Cryptography*. Reading: Addison–Wesley USA.
- Stuart T.E., Podolny J.M. (1996). Local search and the evolution of technological capabilities. *Strategic Management Journal*, Summer Special Issue 17: 21–38.

- Swanson K., McComb D., Smith J., McCubbrey D. (1991). The Application Software Factory. Applying Total Quality Techniques to Systems Development. *MIS Quarterly* 15(4): 567-579.
- Terza J.V. (1998). Estimating count data models with endogenous switching: Sample selection and endogenous treatment effects, *Journal of Econometrics* 8: 129-154.
- Tushman M.L., Anderson P. (1986). Technological discontinuities and organizational environments. *Administrative Science Quarterly* 31(3): 439-465.
- Trispas M., Gavetti G. (2000). Capabilities, Cognition, and Inertia: Evidence from Digital Imaging'. *Strategic Management Journal* 21: 1147-1161.
- Torrise S. (1998). Industrial Organisation and Innovation: an International Study of the Software Industry: Edward Elgar Cheltenham, UK.
- Van den Bosh F.A.J., Volberda H.W., de Boer M. (1999). Co-evolution of Firm Absorptive Capacity and Knowledge Environment: Organizational Forms and Combinative Capabilities. *Organization Science*. 10 (5): 551-568.
- Vella F. (1998). Estimating Models with Sample Selection Bias: A Survey. *Journal of Human Resource*, 33(2): 127-169
- von Hippel E. (1998). Economics of Product Development by Users: The Impact of "Sticky" Local Information. *Management Science* 44 (5): 629-644.
- von Hippel E. (2001). User Toolkits for Innovation. *Journal of Product Innovation Management* 18 (4): 247-257.
- von Hippel E., von Krogh G. (2003). Open Source Software and the "Private-Collective" Innovation Model: Issues for Organization Science. *Organization Science* 14 (2): 209-223.
- Zahra S.A., George G. (2002). Absorptive Capacity: A Review, Re-conceptualization, and Extension. *Academy of Management Review*. 27 (2): 185-203.

Table 1. Expected effects of firm assets on OSS product introduction intensity

	<i>Sector components</i>	
	Software	Hardware
<i>Firm Assets</i>		
Upstream	Favor	?
Downstream	Harm	Favor

Table 2: Sample firms that introduced OSS products between 1980 and 2003 by core sector of activity

	OPSYS Niche		APP Niche	
	Firms	Av. size (Empl.)	Firms	Av. size (Empl.)
Electronics	13	42,827	7	34,840
Hardware	77	11,390	30	24,985
Software	195	1,155	154	1,368
Semiconductors	18	9,451	2	39,434
Telecommunications	12	13,185	11	31,458
Other	5	283,190	9	6,586
Total	320	10,636	213	7,927
N. of products	877		360	

Source: Our elaborations from Infotrac's General Business File ASAP and PROMT database and from Icarus, Amadeus and Jade databases

Table 3: Descriptive Statistics (Means and Standard Errors in parenthesis) for the original sample and the control sample

	P_i=1 (OSS product introduction)		P_i=0 (No OSS product introduction)	
	OPSYS Niche (n=426)	APP Niche (n=213)	OPSYS Niche (n=426)	APP Niche (n=213)
TRADEMARKSOFT	7.031 (1.188)	9.587 (1.754)	6.253 (0.795)	5.897 (0.987)
TRADEMARKHARD	4.637 (0.816)	5.727 (1.209)	2.828 (0.408)	2.286 (.377)
PATENTSOFT	12.406 (4.492)	16.845 (6.707)	1.787 (0.636)	1.319 (.528)
PATENTHARD	16.347 (6.144)	21.760 (9.089)	5.025 (1.540)	3.356 (1.334)
SIZE	10636 (4621.79)	7927.07 (2132.43)	5973.266 (1255.426)	8837.263 (2148.311)
AGE	13.572 (1.008)	14.441 (1.355)	18.709 (1.179)	19.657 (1.566)
TIME2MKT	40.656 (.929)	41.958 (1.421)	-	-
OPSYS-before*	-	0.188 (0.027)	-	-
APP-before*	0.069 (0.014)	-		

* dummy variables

Table 4. Non-linear least squares estimation (*corrected standard errors in parenthesis*)

	OPERATING SYSTEMS		APPLICATIONS	
	(1)	(2)	(1)	(2)
TRADEMARKSOFT	-0.012*** (0.004)	-0.014*** (0.004)	-0.024** (0.010)	-0.037*** (0.010)
TRADEMARKHARD	0.030*** (0.009)	0.032*** (0.009)	0.023 (0.014)	0.029* (0.016)
PATENTSOFT	0.014*** (0.003)	0.014*** (0.003)	0.007 (0.005)	0.009 (0.006)
PATENTHARD	-0.010*** (0.003)	-0.010*** (0.002)	-0.004 (0.003)	-0.006 (0.004)
SIZE	0.000 (0.000)	0.000 (0.000)	0.000** (0.000)	0.000*** (0.000)
TIME2MKT	0.033*** (0.009)	0.034*** (0.009)	0.007*** (0.002)	0.004* (0.002)
OPSYS-before [♦]	-		0.547*** (0.162)	-
APP-before [♦]	-0.322 (0.209)	-	-	-
Firm Sector	Yes	Yes	Yes	Yes
Country	Yes	Yes	Yes	Yes
Beta0	-8.942 (6.622)	-9.817 (6.280)	-0.104 (0.259)	0.370 (0.277)
Theta*	0.409 (0.549)	0.399 (0.562)	-0.464* (0.258)	-0.856*** (0.257)
Log likelihood	-845.387	-845.792	-331.435	-339.422
No. of observations	640	640	426	426

[♦] dummy variables

* The significant estimate for the covariance term θ in the Applications niche reflects the presence of unobserved factors that are common to the two decisions of the firm. Its negative sign shows that an overestimation of the average number of product introductions would have occurred without controlling for selectivity.

Figure 1: Predicted number of product introductions in OPSYS as a function of the pre-entry stock of trademarks (a) and patents (b) in hardware (—) and software (---)

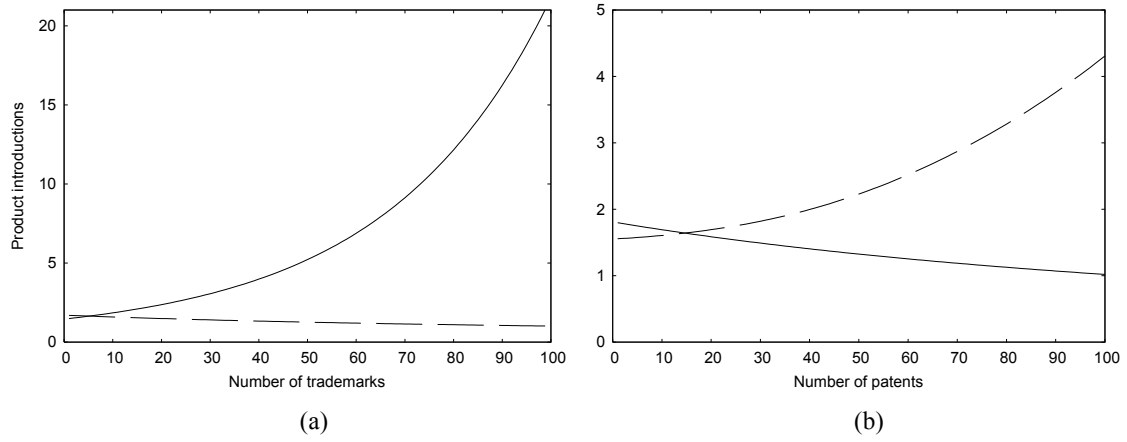
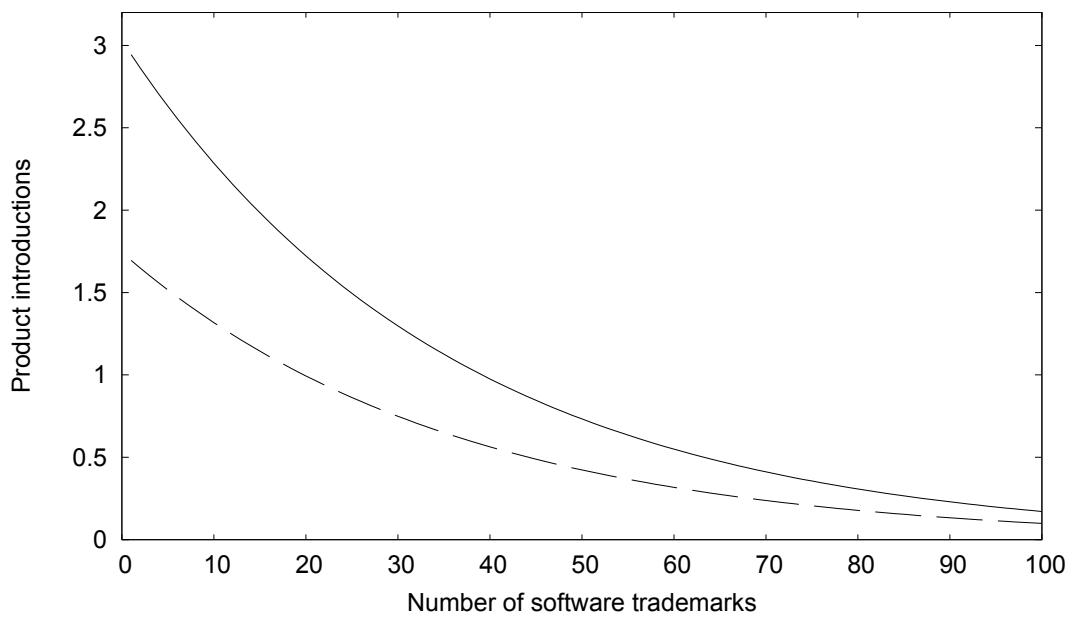


Figure 2: Predicted number of product introductions in APP as a function of the pre-entry stock of software trademarks with OPSYS-BEFORE=1 (—) and OPSYS-BEFORE=0 (---)



Appendix

Search methods and error testing

We employed two search algorithms for patents and trademarks. The general idea of this procedure was to construct algorithms that reduced the probability to fail in identifying a correct hardware or software trademark or patent. For patent we searched in the patent abstract the following string of words: ["computer software" or "operating system" or "computer program" or "software algorithm" or "data processing" or "software application"] for software, and ["computer server" "computer hardware" or "motherboard" or "peripherals" or "workstation" or "mainframe" or "disk driver" or "area network"] for hardware. For the trademarks, we used the same algorithms in the trademark description of goods and services. In order to validate the accuracy of these algorithms, we compared the results of our search against a random sample of 200 patents and trademarks which a software engineer had read and classified into software/no software, hardware/no hardware patents and trademarks. With a conservative approach, we selected the random sample inside the general electronic technological and product classes excluding too distant classes (like food, textile, firearms...for more details see the USPTO technological and product classification, www.uspto.gov). The error percentages are listed below. From 107 non-software patents, this algorithm spotted 23 patents (21.4%) as software patents. From 93 software patents, 8 patents (8.6%) escaped from the algorithm search. Similarly, from 84 non-software trademarks 12 (14.2%) were include as software and from 116 software trademarks 9 (7.7%) escaped. For hardware, from 104 non-hardware patents, we had 16 errors (15.3%) and from 96 hardware patents we had 9 errors (9.3%). From 124 non-hardware trademarks, we had 18 errors (14.5%) and from 76 hardware trademarks we had 6 errors (7.8%). As expected, errors for hardware are less severe than for software and errors in trademarks are less severe than in patents. Overall, the error percentages are similar to Bessen and Hunt (2003) and seem reasonably acceptable.

Table A. *Examples of software/hardware patents and trademarks*

	<i>Software</i>	<i>Hardware</i>
<i>Trademark</i>	Word Mark LANGUAGEWARE Goods and Services: Computer software for use in processing natural language text, namely, an application designed to perform a set of linguistic functions to process natural language... Serial Number 78203799 Filing Date January 15, 2003 Owner International Business Machines Orchard Road Armonk NEW YORK 10504	Word Mark 1350 Goods and Services ... computer hardware; computers; servers; computer connection cabling; computer network hubs, switches and routers; computer networking hardware... Serial Number 78579881 Filing Date March 3, 2005 Owner International Business Machines Orchard Road Armonk NEW YORK 10504
<i>Patent</i>	Title Real-time evaluation of compressed picture quality within a digital video encoder Abstract Method, system and computer program product are provided for real time evaluation of compressed picture quality, in hardware, software or a combination thereof, during encoding of a sequence of video data... Assignee: International Business Machines Corporation (Armonk, NY) Appl. No.: 020904 Filed: February 5, 1998 Patent Number 6,252,905	Title Computer equipment having an earthquake damage protection mechanism Abstract Relatively heavy electrical equipment, such as a computer mainframe or server unit, is provided with a mechanism which prevents and/or mitigates damage to the equipment caused by seismological or other activity... Assignee: International Business Machines Corporation (Armonk, NY) Appl. No.: 457216 Filed: December 9, 1999 Patent Number 6,134,858